

## INTEGRATED MODEM OPERATING WITHOUT NECESSITATING DEDICATED CONTROLLER

**Patent number:** JP2017753  
**Publication date:** 1990-01-22  
**Inventor:** PINDAR ANDREW; MARCROFT DUANE; NICHOLS III ANDREW J  
**Applicant:** NATL SEMICONDUCTOR CORP <NS>  
**Classification:**  
- international: H04M11/00  
- european:  
**Application number:** JP19890100321 19890421  
**Priority number(s):**

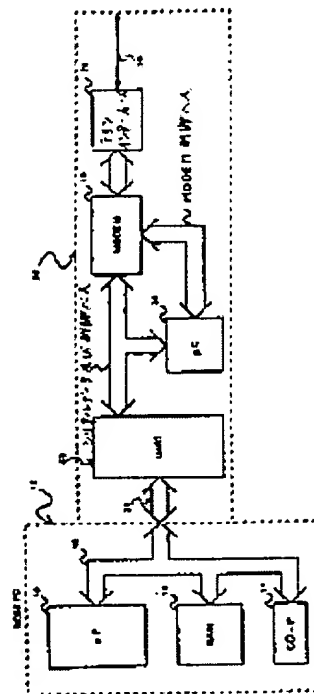
**Also published as:**

EP0340613 (A1)  
US5170470 (A1)  
EP0340613 (B1)

## Abstract of JP2017753

**PURPOSE:** To eliminate the need of any controller dedicated for MODEM by acting a processor as a virtual controller.

**CONSTITUTION:** A host processor 12 transmits characters to a register in an MODEM circuit 18 for emulating a register in an UART 22 under the initial control of a communication program. The MODEM circuit 18 receives those characters, transmits an interruption request to the host processor 12, and it is branched to a controller emulation program, that is, an MODEM driver. The processor 12 retrieves the characters in the processor 12, and processes them under the instruction of this program.



Data supplied from the **esp@cenet** database - Worldwide

## ⑫ 公開特許公報(A) 平2-17753

⑤Int.Cl.<sup>5</sup>

H 04 M 11/00

識別記号

3 0 2

庁内整理番号

8020-5K

⑬公開 平成2年(1990)1月22日

審査請求 未請求 請求項の数 30 (全17頁)

⑭発明の名称 専用コントローラ無しで動作する集積化MODEM

⑰特 願 平1-100321

⑱出 願 平1(1989)4月21日

優先権主張 ⑲1988年5月2日⑳米国(US)㉑189,062

⑳発 明 者 アンドリュー ビンダ アメリカ合衆国, カリフォルニア 94086, サニーベル、  
ジャクソン アベニュー 206㉑発 明 者 デュアン マークロフ アメリカ合衆国, カリフォルニア 94070, サン カルロ  
ス, ジュニーバ アベニュー 1368㉒出 願 人 ナショナル セミコン アメリカ合衆国, カリフォルニア 95052, サンタ クラ  
ダクタ コーポレーシ  
ョン

㉓代 理 人 弁理士 小橋 一男 外1名

最終頁に続く

## 明 細 書

## 1. 発明の名称

専用コントローラ無しで動作する集積化  
MODEM

## 2. 特許請求の範囲

1. 他のコンピュータと通信する為に使用する内部バス上で制御文字及びデータ文字を発生させ且つ受け取る為にプログラムメモリ内の通信プログラムに回答するホストプロセサを持ったコンピュータにおいて、前記ホストプロセサを仮想コントローラとして使用する通信方式において、前記内部バスからデータ文字を受け取り且つ該データ文字を電話線へ送信し且つ前記電話線からデータ文字を受け取り且つ該データ文字を前記内部バスへ供給する手段を具備するMODEM、前記通信プログラムに回答して前記ホストプロセサによって発生される制御文字及びデータ文字を受け取り且つ前記文字の受領を表す所定の割込要求を発生させる少なくとも1個のレジスタを具備するインターフェース手段、前記プログラムメモリ内に

格納されており且つ前記MODEMの動作に関連する命令を収容するMODEMドライバプログラム、前記ホストプロセサをして前記プログラムメモリ内に格納されている前記MODEMドライバプログラムにアクセスさせ且つ前記MODEMドライバプログラム内に収容されている命令に従って前記MODEMを制御させる為に前記インターフェース手段により前記所定の割込要求の発生に回答する制御手段、を有することを特徴とする通信方式。

2. 特許請求の範囲第1項において、前記MODEMドライバプログラムは、前記ホストプロセサ用のオペレーティングシステムの一部を有していることを特徴とする通信方式。

3. 特許請求の範囲第1項において、前記MODEMドライバプログラムは、ターミネイトアンドステイ常駐プログラムとして前記プログラムメモリ内に格納されているアプリケーションプログラムを有していることを特徴とする通信方式。

4. 特許請求の範囲第1項において、前記プ

ログラムメモリは、夫々の割込要求に対してのアドレスを収容するテーブルを具備しており、且つ前記制御手段は、特定の割込要求と関連する前記テーブル内のアドレスを前記MODEMドライバプログラムに関連する異なったアドレスと置換させる為の手段を有することを特徴とする通信方式。

5. 特許請求の範囲第4項において、前記コンピュータは周期的な割合でタイミング割込要求を発生し、且つ前記制御手段は前記特定の割込要求に対して前記テーブル内に格納されているアドレス検査し且つ前記格納アドレスが前記異なったアドレスと同一ではない場合にそれを前記異なったアドレスと置換させるべく各タイミング割込要求に応答することを特徴とする通信方式。

6. 特許請求の範囲第1項において、前記インターフェース手段は、更に、前記ホストプロセサ用に意図された文字を格納する第2レジスタであって前記第2レジスタからの文字の検索を表す第2割込要求を発生させる第2レジスタを有しており、且つ前記制御手段も前記第2割込要求に応

答して前記MODEMドライバプログラムの制御下において前記受領した情報に従って前記情報を処理すると共に前記MODEMを制御する、上記各ステップを有することを特徴とする方法。

9. 特許請求の範囲第8項において、前記ホストプロセサの制御を転送するステップにおいて、割込要求アドレステーブルの修正を行い、通信割込要求が発生されると、前記MODEMドライバプログラムに関連するメモリ位置を前記ホストプロセサをしてアドレスさせるポインタを具備することを特徴とする方法。

10. 特許請求の範囲第9項において、前記テーブルの修正が、前記通信割込要求に関連するアドレスポインタが前記MODEMドライバプログラムに関連するもの以外のメモリ位置を示しているか否かを決定する為に前記テーブルを検査し、それが前記MODEMドライバプログラムと関連していない場合に前記他のメモリ位置を格納し、且つ前記テーブル内の前記他のメモリ位置を前記MODEMドライバプログラムと関連する前記メ

モリ位置と置換させる、上記各ステップを有することを特徴とする方法。

7. 特許請求の範囲第1項において、前記コンピュータは、前記ホストプロセサがその上に装着されている親基板を具備しており、且つ前記MODEM及び前記レジスタは前記親基板上に装着されている集積回路チップ内に収容されていることを特徴とする通信方式。

8. コンピュータ内においてMODEMを制御する方法において、コンピュータと通信リンクとの間でのデータの転送に関する命令を発生させる為に通信プログラムの制御下でコンピュータのホストプロセサを動作させ、前記MODEMによって何時情報が受領されたかを表す為に前記ホストプロセサへ割込要求を発生し、前記割込要求の発生に応答して前記通信プログラムからMODEMドライバプログラムへホストプロセサの制御を転送し、前記ホストプロセサにおいて前記MODEMにおいて受領された情報を検索し、前記MO

DEMODドライバプログラムの制御下において前記受領した情報に従って前記情報を処理すると共に前記MODEMを制御する、上記各ステップを有することを特徴とする方法。

11. 特許請求の範囲第10項において、更に、前記情報処理ステップの終了後に前記ホストプロセサの制御を前記通信プログラムへ復帰させるステップを有することを特徴とする方法。

12. 特許請求の範囲第11項において、前記ホストプロセサは、前記プロセサをして前記テーブルから格納されている前記他のメモリ位置をアドレスさせることによって前記通信プログラムへ復帰されることを特徴とする方法。

13. 特許請求の範囲第10項において、前記コンピュータは、周期的間隔でタイミング割込要求を発生し、且つ前記テーブルを検査するステップは各タイミング割込要求の発生によって実行されることを特徴とする方法。

14. 特許請求の範囲第8項において、前記コンピュータは、周期的間隔でタイミング割込要求を発生し、且つ各タイミング割込要求の発生も前記ホストプロセサの制御を前記MODEMドライ

バプログラムへ転送させることを特徴とする方法。

15. 特許請求の範囲第14項において、前記MODEMドライバプログラムが、前記プロセサの制御を前記通信プログラムへ復帰させた時にそれがある動作状態の表示を格納し且つ各タイミング割込要求の発生後前記状態へ復帰することを特徴とする方法。

16. 通信リンクとホストプロセサを持ったコンピュータとの間で情報を転送するMODEM回路において、前記ホストプロセサから前記通信リンクへ情報を送信する変調器、前記通信リンク上の情報を受領する復調器、前記ホストプロセサによって発生された文字を受領し且つ該文字を前記変調器へ供給する送信保持レジスタ、前記ホストプロセサによって前記送信保持レジスタ内に文字が配置された場合に前記ホストプロセサへ第1割込要求を発生する手段、前記復調器から前記ホストプロセサへ供給されるべき文字を受信する受信保持レジスタ、前記ホストプロセサが前記受信保持レジスタから文字を検索する場合に前記ホス

信保持レジスタ内に配置させる手段を有すること  
特徴とするMODEM回路。

19. 特許請求の範囲第18項において、前記制御手段は前記第1割込要求に応答して前記送信保持レジスタから文字を検索し且つ前記第3割込要求に  
応答して前記検索した文字を前記送信中間レジスタ内に配置させ、その際に非同期対同期バッファの必要性無しで前記ホストプロセサと前記変調器との間でデータを転送させることを特徴とするMODEM回路。

20. 特許請求の範囲第18項において、前記制御手段は前記第2割込要求に  
応答して前記受信中間レジスタから文字を検索し且つ前記第4割込要求に  
応答して前記検索した文字を前記受信保持レジスタ内に配置させ、その際に同期対非同期バッファの必要性無しで前記ホストプロセサと前記復調器との間でデータを転送することを特徴とするMODEM回路。

21. 特許請求の範囲第16項において、更に、データが前記通信リンク上を伝送される場合に前

トプロセサへ第2割込要求を発生する手段、を有することを特徴とするMODEM回路。

17. 特許請求の範囲第16項において、更に、前記変調器へ文字を供給する送信中間レジスタ、前記送信中間レジスタから文字が除去された時に前記ホストプロセサへ第3割込要求を発生する手段、前記復調器からの文字を受信する為の受信中間レジスタ、前記受信中間レジスタ内に文字が配置される場合に前記ホストプロセサへ第4割込要求を発生する手段、を有することを特徴とするMODEM回路。

18. 特許請求の範囲第17項において、更に、前記ホストプロセサをして前記送信保持レジスタ内の文字を検索させ且つ前記文字が前記通信リンク上を通信されるべきデータであることを表す場合に該文字を前記送信中間レジスタ内に配置させるべく前記ホストプロセサを制御すると共に、文字が前記通信リンクを介して受信されたデータを表す場合に前記ホストプロセサをして前記受信中間レジスタから文字を検索し且つ該文字を前記受

信保持レジスタと前記夫々の変調器又は復調器との間にデータ経路を選択的に確立し且つ伝送したデータ以外の情報が前記MODEM回路と前記ホストプロセサとの間で交換される場合に前記データ経路をインタラプトする制御手段を有すること  
特徴とするMODEM回路。

22. コンピュータにおけるMODEMを制御する方法において、コンピュータと通信リンクとの間でのデータの伝送に関する命令を発生する為に通信プログラムの制御下で前記コンピュータのホストプロセサを動作させ、何時情報が前記MODEMによって受信されたかを表す為に前記ホストプロセサへ割込要求を発生し、前記割込要求の発生に  
応答して前記通信プログラムからMODEMドライバプログラムへ前記ホストプロセサの制御を転送し、前記MODEMドライバプログラムの制御下において前記MODEMによって受信された情報が前記通信プログラムによって又は前記MODEMドライバプログラムによってのサービスを必要とするか否かを決定し、前記情報が前

記通信プログラムによってサービスされるべきものである場合に前記ホストプロセサの制御を前記通信プログラムへ復帰させ、前記情報が前記ドライバプログラムによってサービスされるべきものである場合に前記ドライバプログラムによる前記ホストプロセサの制御を維持し且つ前記ホストプロセサにおける前記情報を検索する、上記各ステップを有することを特徴とする方法。

23. 特許請求の範囲第8項において、前記ホストプロセサの制御を転送するステップが、通信割込要求が発生される場合に前記MODEMドライバプログラムと関連するメモリ位置を前記ホストプロセサをしてアドレスさせるポインタを包含すべく割込要求アドレステーブルの修正を包含することを特徴とする方法。

24. 特許請求の範囲第23項において、前記テーブルの修正は、前記通信割込要求に関連するアドレスポインタが前記MODEMドライバプログラムに関連するもの以外のメモリ位置を参照するかどうかを決定する為に前記テーブルを検査

イバプログラムへ転送させることを特徴とする方法。

28. 特許請求の範囲第27項において、前記MODEMドライバプログラムは、前記プロセサの制御を前記通信プログラムへ復帰させるとそれが存在する動作状態の表示を格納し且つ各タイミング割込要求の発生後に前記状態へ復帰することを特徴とする方法。

29. 特許請求の範囲第22項において、前記MODEMドライバプログラムは前記ホストプロセサ用のメインメモリ内に格納されることを特徴とする方法。

30. 特許請求の範囲第29項において、前記MODEMドライバプログラムはターミネイトアンドステイ常駐プログラムとして格納されることを特徴とする方法。

### 3. 発明の詳細な説明

#### 技術分野

本発明は、大略、電話線又はその他の通信リンクによってコンピュータ間でデータを転送するこ

し、それが前記MODEMドライバプログラムと関連するものでない場合には前記他のメモリ位置を格納し、且つ前記テーブル内の前記他のメモリ位置を前記MODEMドライバプログラムに関連する前記メモリ位置と置換させる、上記各ステップを有することを特徴とする方法。

25. 特許請求の範囲第24項において、前記ホストプロセサは、前記プロセサをして前記テーブルから格納されている前記他のメモリ位置をアドレスさせることにより前記通信プログラムへ復帰させることを特徴とする方法。

26. 特許請求の範囲第24項において、前記コンピュータは周期的な間隔でタイミング割込要求を発生し、且つ前記テーブルを検査するステップは各タイミング割込要求の発生後に実行されることを特徴とする方法。

27. 特許請求の範囲第22項において、前記コンピュータは周期的な間隔でタイミング割込要求を発生し、且つ各タイミング割込要求の発生も前記ホストプロセサの制御を前記MODEMドラ

とを可能とするMODEMに関するものである。更に詳細には、本発明は、ソフトウェアの修正無しで現在市販されている通信アプリケーションソフトウェアの制御下で動作することを可能とする為にコンピュータの構成内に組み込むか又は埋め込むべく適合されており且つ周辺に配設されたMODEMの機能をエミュレートするMODEMを具備する通信方式に関するものである。特にそれに限定するわけではないが、本発明は、特に「パーソナルコンピュータ」としても知られるマイクロコンピュータに使用すべく構成された通信方式に関するものである。

#### 従来技術

コンピュータ間の通信においては、2つのタイプの情報が発生され、即ちコンピュータ間で転送されるべきデータ自身と、データを適切に送信及び受信するべくMODEMを構成する制御文字(コントロールキャラクタ)である。データは多数の異なる発生源から送られ、例えば直接的にコンピュータのキーボードから、又はディスクに

格納されているファイルから送給され、且つ制御文字は、通信プログラムの制御下でホストプロセサによって発生される。データが電話線を介して送られる一方、制御文字は電話線によって担持されることは意図されておらず、従ってMODEMはこれらの制御文字がホストプロセサによって発生される場合にそれらの制御文字を認識し且つ解釈することが必要である。MODEM用の1つの良く使用されているものは、制御文字を解釈し且つ処理する為の専用コントローラを具備する8250UART (ユニバーサル非同期受信器/送信器)を組み込んだものである。

動作に付いて説明すると、コンピュータのホストプロセサが例えば特定の番号をダイヤルする等の機能を実施する為にMODEMへの命令を発生すると、それは8250UARTのレジスタ内に制御文字を配置させる。コントローラはUARTから命令文字を検索し且つMODEMが電話のダイヤリングに対応するパルス又はトーン信号を送信すべく構成させる。次いでプロセサは、UAR

アドレスジャンプさせて該文字を受け取る。その代わりに、又は付加的に、UARTは適宜のレジスタ内にステータスフラッグをセットして、プロセサによって受け取られるべく文字が所定の位置に存在することを表す。プロセサは、ステータスレジスタを周期的にポーリングして、ステータスフラッグがセットされているか否かを決定することが可能である。ステータスフラッグの検知又は割込に応答して、プロセサは接続が確立されたことを表す応答文字を受け取る。次いで、それは送信されるべきデータを検索するか又はそうでなければ発生し、且つMODEMコントローラはこのデータを、電話線を介して送信する為に、それを中断するのではなく、MODEMへ直接的に送給することを可能とさせる。

過去において、専用コントローラを具備するこのタイプのMODEMは、マイクロコンピュータ用の周辺装置として存在していた。例えば、それらはコンピュータの外部に位置させる場合があり、且つコンピュータ上に位置されたシリアル通信ポ

T及びコントローラを介して、MODEMヘダイヤルすべき番号を供給する。該番号のダイヤリングが完了した後、コントローラは発呼されたMODEMからの応答を待つ。発呼されたMODEMが応答し且つ電話線を介して接続が確立されると、コントローラはホストプロセサへ応答文字を送給して、接続が確立され且つデータの送信を実施することが可能であることを表す。

コントローラがMODEMをコンフィギュア即ち特定の形態に構成させ且つ接続が確立される間に、通信プログラムはホストプロセサの制御を放棄してプロセサがそのメインプログラムを実行し且つその他の内部タスクを行うことを可能とさせることが可能である。従って、MODEMが接続が確立されたということをプロセサへ通知する為に応答文字を送信する準備がなされると、通信プログラムに該文字を受け取らせるプロンプトを与えることが必要である。1つのタイプのプロンプトは、UARTによって発生される割込要求であり、それはプロセサを通信プログラム中の適宜の

ートによってアクセスされる場合があった。この場合、ホストプロセサとMODEMとの間のデータ及び制御文字の転送は、シリアルな態様で行われる。一方、MODEMはコンピュータ自身の為のハウジング内に組み込むことが可能である。典型的に、コントローラ及び他のインターフェース回路を具備するMODEMは、コンピュータ用の親基板上的拡張スロット内にプラグ接続されるプリント回路基板、又はカード等の上に装着される場合がある。この形態において、ホストプロセサ及びMODEMの間の通信は、コンピュータ用のデータ及び制御バスを直接使用して、パラレルな態様で行うことが可能である。然し乍ら、MODEMは、ホストプロセサにより直接アドレス可能であるのではなく、通信ポートによりアクセスされるものである。

MODEMを周辺装置として設けるのではなく、コンピュータの構成の中にMODEMを直接組み込むことが望ましい。特に、MODEM回路をコンピュータの親基板上に直接設け且つMODEM

にI/Oポートを介してアクセスする必要のない独特の装置アドレスを与えることが望ましい。更に、別体の専用コントローラを使用する代わりに、MODEM用のコントローラとしてコンピュータのホストプロセサを使用することが望ましい。

コンピュータの構成の中にMODEMを組み込むことは、消費者の観点から、この様な特徴の全体的な所望性に影響を与えることのある或る種の実際的な考察を必要とする。MODEMがコンピュータアーキテクチャの一部となると、典型的なアプローチは、通信プログラムでホストプロセサ用のオペレーティングシステムを介してMODEMを直接的に制御させる場合であるかもしれない。従って、このタイプの適用の為に構成された通信プログラムは、オペレーティングシステム用に特別に書かれる必要がある。従って、パーソナルコンピュータにおいて使用される異なったMODEMアーキテクチャの為に、ポピュラーなパーソナルコンピュータ市場に役立つことが可能であるとは思われない。

#### 目 的

本発明は、以上の点に鑑みなされたものであって、上述した如き従来技術の欠点を解消し、電話線又はその他の通信線を使用してコンピュータ間でデータを転送させることの可能なMODEMを提供することを目的とする。

#### 構 成

専用コントローラを必要とせず、この様なコントローラ及び該コントローラ用のUART又は同様のインターフェースを具備するMODEM用に設計された既存の通信ソフトウェアで動作可能なコンピュータ用の埋込型MODEMを提供することが特に望ましい。

本発明によれば、常駐ドライバプログラムの指示の下で、MODEMコントローラとしてコンピュータのホストプロセサを使用することを特徴としている。動作に付いて説明すると、通信プログラムの初期制御下でホストプロセサが、UART内のレジスタをエミュレートするMODEM回路内のレジスタへ文字を送給する。これらの文字を

更に詳細に説明すると、パーソナルコンピュータにおいて通信機能を与える為の現在入手可能なアプリケーションプログラムは、専用コントローラ及び8250UART又はこの様なUARTの機能をエミュレートする装置を具備するMODEMで動作すべく構成されている。埋込型MODEMを具備しているがこの様なインターフェースユニットを組み込むことのないコンピュータは、既存のパーソナルコンピュータアーキテクチャ用に書かれている現在入手可能な通信プログラムを使用することが可能なものではない。その結果、この様なコンピュータは、新たなソフトウェアを必要とし、従って消費者にとって魅力的なものではない。埋込型MODEMを動作させる為に新たな通信プログラムを与えることが必要であると、この付加的なプログラムを書き且つ入手する為のコストはコンピュータの正味費用を増加させ、且つこの新たなプログラムを動作させる為の新たなプロトコル及び命令をユーザが学習することを必要とする場合がある。

受け取ると、MODEMは割込要求をホストプロセサへ送り、それはコントローラエミュレーションプログラム、即ちMODEMドライバへ分岐させる。このプログラムの指示の下において、プロセサは該レジスタ内の文字を検索し且つそれを処理する。例えば、該文字が送信すべきデータである場合、該文字をMODEM内の別のレジスタへ与えることが可能であり、該レジスタは該文字をMODEM変調器及び送信器へ送給する。該文字が命令文字又は制御文字である場合、該文字はドライバによって解釈され且つ特定の動作の為にMODEMを構成する為に使用される。

例えば、動作の完了を示す為にドライバが通信プログラムへ送給すべき情報を持っている場合、該ドライバは、UARTの機能をエミュレートするMODEM回路の別のレジスタ内に文字を配位させ、次いで通信プログラムに対してのホストプロセサの制御を放棄する。この通信プログラムが通常の態様でこの文字を検索すると、割込要求がホストプロセサへ与えられて、該レジスタが空で

あることを表す。この割込要求を受け取ると、プロセサはドライバへ復帰して、次の文字をレジスタ内に配置させるか又は通信プログラムによる該文字の検索に対して適切なその他の動作を行う。

従って、プロセサは仮想コントローラとして機能する。即ち、通信プログラムは該制御文字を送らせるか、又は外部コントローラとして解釈されるべき装置から検索させる。然し乍ら、事実上、これらの文字は、ドライバプログラムに従って、ホストプロセサへ読み戻されるか又はそれによって発生される。その結果、MODEMは、別の専用コントローラの必要性無しで動作することが可能である。

本発明が前述した動作を実施する為に機能する詳細な態様、本発明の別の特徴、及び本発明の利点等は、以下の好適実施例に関しての詳細な説明において説明する。

#### 実施例

本発明の理解を容易とする為に、第1図を参照して典型的なMODEM構成に関して説明する。

られるデータに対しては直列／並列変換器として機能する。UART 22、ホストプロセサ10、及びMODEM 18間のデータ転送は、専用マイクロコントローラ26の制御下で行われる。更に、マイクロコントローラ26は、ホストプロセサ10から送信される制御文字に従って命令を実行するべくMODEM 18をコンフィグャー即ち構成する。ラインインターフェース回路28が、MODEM 18を従来の電話線30へ接続させる。このラインインターフェース回路28は、MODEMと電話線との間で転送されるデータを適宜バッファし、且つリング検知及びパルスダイアリング等のその他の機能を実行する。

UARTの代わりに、USART(ユニバーサル同期／非同期受信器／送信器)を使用することも可能である。この様な構成は、同期エラー補正及びデータ圧縮プロトコルをデータ転送の為に使用される場合に特に望ましい場合がある。以下の説明は、USARTを有するMODEMに関してのものであるが、本発明は、USART又はその

MODEMが接続されるコンピュータは、屢々「親基板(マザーボード)」として呼称されるプリント回路基板12上に装着されているホストマイクロプロセサ10を有している。プロセサ10は、内部並列バス16によって該親基板上に装着されているその他のアドレス可能な装置14と通信する。例えば、これらの装置は、メインメモリシステム(RAM)、コプロセサ等を有することが可能である。

第1図に示したコンフィギュレーション即ち構成において、MODEM 18は、親基板12内にプラグ接続される別体のプリント回路基板20上に装着されている。MODEM 18は、UART 22によってホストプロセサ10と通信し、且つコンピュータ用の通信ポート、例えばCOM1によってアクセスされる。UARTはコンピュータの内部バス16と直列通信する並列バス24へ接続されている。UART 22は、主に、プロセサ10から送られるデータに対して並列／直列変換器として機能し、且つMODEM 18から受け取

他の通常の通信コントローラを使用するMODEMへも同様に適用可能であることを理解すべきである。

MODEMを親基板12へプラグ接続するカード上に位置させる代わりに、MODEMをコンピュータハウジング外部の構成体内に設けることが可能である。この様な場合、コンピュータとMODEMとの間の情報の転送は、ハードウェア通信ポートを介してアクセスされる例えばRS-232インターフェース装置等のシリアルインターフェースにより行うことも可能である。このタイプの構成においては、UARTの機能は、コンピュータハウジング内に位置されているシリアルI/Oカードにより与えられる。

MODEM 18、UART 22、及びマイクロコントローラ26は第1図において別々の構成体として示されているが、実際の構成においては、これらの回路の任意の2つ、又はそれらの全てを単一の集積回路チップ内に組み込むことが可能である。本発明の好適実施例においては、MODEM



M回路は単一のチップ上で、該チップをUART 22の機能をエミュレートすることを可能とさせるレジスタと結合されている。

本発明によれば、MODEMコントローラの機能は、第1図に示した如く、専用コントローラ26によるのではなく、ホストプロセサ10によって実施される。その結果、MODEM回路は、第2図に示した如く、親基板12上に位置させることが可能である。この構成の場合、制御及びデータワードは、プロセサ10とMODEM18との間に、ハードウェアインターフェースとして機能する通信ポートに対する必要性無しに、コンピュータの内部バスのみを使用して、並列フォーマットでプロセサ10とMODEM18との間で直接交換させることが可能である。使用されるハードウェアインターフェースは、ラインインターフェース回路28のみであり、該回路も該親基板上又は別のカード53上に装着させることが可能である。

このタイプの構成でMODEMを動作させる為

に、ホストプロセサ10の制御は、通信プログラムとコントローラエミュレーションプログラム乃至は装置ドライバとの間で交互に行われる。これら2つのプログラムの間でのスイッチングは、通信割込要求によって行われる。この本発明の基礎となる概念を、第3図に示したメモリマップを参照して更に詳細に説明する。

第3図を参照すると、コンピュータのメインメモリは、入出力システム54、オペレーティングシステム56、及び1個以上のアプリケーションプログラムを所定のアドレスにおいて収納している。これらのアプリケーションプログラムの中において、通信プログラム58は、MODEMを介してデータを送信し且つ受信する為にプロセサを制御する。メインメモリ内には、更に割込ベクトルアドレステーブル60も格納されている。このテーブルは、システム内で発生させることの可能な各タイプの割込要求に対するアドレスを有している。割込要求の受領に応答して、プロセサはテーブル60を参照して、それが該インタラプト即

ち割込をサービスする為に進行するメモリアドレスを決定する。

従来の適用において、通信割込用のテーブル60内に格納されているアドレスが通信プログラム58を参照することがある。然し乍ら、本発明によれば、テーブル60内の或るアドレスは修正され、従って通信割込はプロセサ10をしてメモリ内の別のアプリケーションプログラム62を参照させる。更に詳細に説明すると、このプログラムは、ターミネイトアンドステイ常駐(terminate-and-stay resident、即ちTSR)MODEMドライバである。その機能は、第4図に示した構成を参照して詳細に説明する。

更に、新しいタイプの通信割込要求は、ドライバが適切に通信動作を制御することを可能とする為に設けられている。基本的に、これらの新しい割込要求は、通常専用コントローラによってサービスされるMODEM内に現存する条件を表す。本発明においては、この様なコントローラの機能はホストプロセサによって実施されるので、これ

らの新たな割込は、コントローラの機能をエミュレートする動作を実施せねばならないことをホストプロセサへ警告する為に設けられている。例えば、コントローラ用に意図されているUARTレジスタ内に文字が配置される毎に、割込要求が発生されて、ホストプロセサをMODEMドライバ62へジャンプさせ且つ該文字を検索させる。動作について説明すると、全ての通信割込はホストプロセサをしてMODEMドライバへ分岐させる。該ドライバは、割込みが通信プログラムによってサービスされる従来の条件の結果であるか又は新たな条件であるかを決定する迄、一時的に、ホストプロセサが該割込に関して動作することを防止する。前者の場合、ドライバは通常のサービスを行う為に、通信プログラムに対するホストの制御を解除する。一方、新しいタイプの割込条件である場合、ドライバは制御を維持し且つ割込要求をサービスする。この動作の更に詳細な説明は、第4図に示した特定のMODEM回路を参照して行う。

第4図を参照すると、本発明の動作において機能的なMODEMチップの部分を示してある。MODEMは2つの基本モードで動作し、即ちデータモード及び命令モードで動作する。データモードの場合、データは電話線を介しての通信の為に、通信プログラムとMODEMとの間で送給される。命令モードの場合、制御文字は通信プログラムからMODEMは又はその逆に送られて、MODEMを実施すべき動作を実行するべく構成させる。データモードで機能する為に、MODEMは変調器32を有しており、それはホストプロセサによって発生されるデータを受け付け且つそれを音声グレード電話線又はその他の形態の通信リンクを介して通信するのに適した適宜のオーディオ周波数信号へ変換させる。同様に、それは、復調器34を有しており、それは通信リンクを介して送信される信号を受け取り且つ該信号をホストプロセサへ供給される適宜のデジタル信号へ変換する。

変調器32及び復調器34は、MODEM回路内の2組のレジスタを介してホストプロセサと通

信する。第4図に示した上側組のレジスタ33は、UART内に通常存在するレジスタをエミュレートする。基本的に、これらは通信プログラムによってアクセスされるレジスタであり、且つこれらのレジスタのみのアドレスは通信プログラムのアドレスフィールド内にある。下側組のレジスタ35は、MODEM内の種々の装置の制御、TSRMODEMドライバ62とのインターフェース、及びそれに関連する割込処理に関係している。例えば、下側組35内のレジスタの1つは、同期ブロックモードにおける動作の同期文字でプリセットすることが可能である。入力ワードはこのレジスタ内に格納される文字と比較され、且つマッチング即ち整合が検知されると、割込を発生して、同期文字がデータブロックの始めにおいて検知されたことを表すことが可能である。

MODEMドライバは、両方の組33及び35内のレジスタへのアクセスを持っている。これらの種々のレジスタは、MODEMを収納するICパッケージ上で使用可能な接続ピンの数に依存し

て、2つの異なった態様の1つでアドレスすることが可能である。適切な数の使用可能なアドレスラインがある場合、各レジスタの直接アドレス動作が望ましい。例えば、両方の組33及び35において全部で32個のレジスタがある場合、5個のアドレスラインを設けて、各レジスタをそのレジスタに独特の二進アドレスで直接アドレスすることが可能である。

然し乍ら、アドレスライン数が制限されている場合、バンク選択技術を使用することが可能である。例えば、通常通信プログラムによってアクセスされるUARTの8個のレジスタをアドレスする為に3本のアドレスラインのみが使用可能である場合、該レジスタを各々が8個のレジスタからなる4個のバンクに分割することが可能である。これらのレジスタバンク中で選択を与える為に、組33のレジスタの1つの中の2個のビット位置を選択したバンクの識別乃至は同定の為に指定することが可能である。例えば、UARTレジスタの組の中に通常見つけられる1個のレジスタは割

込識別(同定)レジスタ31である。通常、このレジスタは、リードオンリ即ち読取専用のレジスタである。然し乍ら、本発明においては、このレジスタ内の任意の数の使用可能なビット位置に、読取/書込能力を与えることが可能である。読取/書込能力を持ったビットは、3本のアドレスライン上に表われるアドレスによってアドレスされるべきレジスタの特定のバンクのレジスタを識別する為に使用することが可能である。4個のバンクのレジスタのみが使用可能である場合、選択したバンクの二進表示を与える為に、割込識別レジスタ31において使用可能なものから単に2ビットのみを使用することが必要であるに過ぎない。

ここのレジスタの機能を、MODEMにおいて実施される特定の動作に関連して説明する。最初にデータモードにおけるMODEMの動作を参照すると、プロセサを制御する通信プログラムが電話線30を介して通信すべきデータ文字を有する場合、それはMODEMレジスタ組33内の送信保持レジスタ36が空であることの表示を待つ。

該レジスタが空であることの表示は、割込要求、又はプロセサによってポーリングされるラインステータスレジスタ39内のビットの設定によって与えることが可能である。ホストプロセサ内の通信プログラムがこの表示を受け取ると、それは該文字を送信保持レジスタ36内に配置させる。レジスタ36内への文字の配置に应答して、MODEM回路は、ホスト/UARTインターフェースにおいて割込ピンを設定することによりホストプロセサへ割込を送り、該送信保持レジスタがフル即ち満杯であることを表す。この割込条件は、下側組35内のレジスタ43の1つの中に特定ビットTXHRFを設定することにより識別される。

該割込がホストプロセサによって受け取られると、それはTSRMODEMドライバ62へ分岐する。該ドライバは、最初に、レジスタ43を検査し且つその割込を発生した条件を決定する。この特定の条件は該ドライバによるサービスを必要とするものであるから、それは該プロセサの制御を維持する。特に、該ドライバは送信保持レジ

スタ36内のビットRXHREがこのレジスタが空であることを表すべくセットされたことを検知すると、受信保持レジスタ42内に該文字を書き込む。該文字がレジスタ42内にエンターされると、MODEM回路はプロセサへ表示、例えば割込み又はステータスビット、を送り、通信プログラムをして該文字を読み取らせる。

命令モードにおいて、制御文字は、通信プログラムの制御下でホストプロセサ10によって発生され、且つMODEMへ送られて、特定の動作を実行すべくそれを構成する。これらの文字は、電話線30を介して送信されるべく意図されていない。それらは、特定の動作を実行する為にMODEMコントローラによって使用される。

MODEMが電話番号をダイヤルし次いで別のコンピュータとの電話接続が確立されるとデータの伝送を開始するようにホストプロセサ10が命令を与える場合の例について説明する。動作を開始する為に、コンピュータのメインメモリ14内に格納されている通信プログラム58がホストプ

ロセサに命令を与えて、最初の文字を、送信保持レジスタ36内の命令ストリング、例えば「A」内に配置させる。TXHRF割込を受け取ると、プロセサはTSRドライバへ分岐して、情報がそれへ送られることを認識し、且つレジスタ36内に格納される文字を受け入れる。この文字は、レジスタ又はバッファ、例えばホストプロセサ用のRAM内の指定したワークスペース即ち作業空間内に格納させることが可能である。プロセサによるレジスタ36からの該文字の検索は、ステータスレジスタ39におけるステータスフラグをセットさせて、該文字が受け入れられたことを表す。この動作は、更に、該レジスタが空であることのホストプロセサ10への表示となる。この表示を受け取ると、命令プログラムは、次の制御文字をデータレジスタ内に配置させる。この処理は、全命令ストリングが受け取られ且つMODEMドライバによって格納される迄継続される。該命令を受け取ると、この場合は電話番号をダイヤルすべき命令であるが、ドライバは1つ又はそれ以上の

タ36からの文字を読取、且つ、それは送信すべきデータであるから、このレジスタが空であることを表す別の割込を受け取ると、それを送信中間レジスタ37へ書き込む。この割込条件は、レジスタ43内の別のビットTXIEの設定によって表される。ドライバによってレジスタ37内に配置された文字は、逐次的に、データバッファ38へ転送され、そこから、該文字はシリアルに変調器32へ転送されて、電話線を介して送信される。同様に、電話線を介して受け取られるデータは、復調器34内において復調され且つバッファ40内に格納される。このバッファから、該データは中間受信レジスタ41内へラッチされる。該MODEM回路は、割込をプロセサへ送給し且つレジスタ43において別のビット位置RXIFをセットしそのレジスタ41が満杯であることを表す。この割込みに応答して、TSRドライバは該レジスタ内の文字を読み取る。この文字は通信プログラムへ送られるべきデータに関するものであるから、ドライバは、割込を受け取り且つレジスタ4

ロセサに命令を与えて、最初の文字を、送信保持レジスタ36内の命令ストリング、例えば「A」内に配置させる。TXHRF割込を受け取ると、プロセサはTSRドライバへ分岐して、情報がそれへ送られることを認識し、且つレジスタ36内に格納される文字を受け入れる。この文字は、レジスタ又はバッファ、例えばホストプロセサ用のRAM内の指定したワークスペース即ち作業空間内に格納させることが可能である。プロセサによるレジスタ36からの該文字の検索は、ステータスレジスタ39におけるステータスフラグをセットさせて、該文字が受け入れられたことを表す。この動作は、更に、該レジスタが空であることのホストプロセサ10への表示となる。この表示を受け取ると、命令プログラムは、次の制御文字をデータレジスタ内に配置させる。この処理は、全命令ストリングが受け取られ且つMODEMドライバによって格納される迄継続される。該命令を受け取ると、この場合は電話番号をダイヤルすべき命令であるが、ドライバは1つ又はそれ以上の

レジスタをセットし且つ装置制御レジスタ50のバンク内の1つ又はそれ以上のレジスタを検査し、それはMODEMを特定の態様で動作させて番号をダイヤルさせ且つ接続を確立する。従って、ドライバはレジスタ50内に情報を配置させ、それはMODEMをパルス又はトーンダイヤルモードへ移行させる。更に、従来のMODEMにおける如く、電話番号がMODEMドライバへ送られ且つRAM内に設けられている適宜のワークスペース内に格納される。この番号は、次いで、MODEM制御レジスタの使用を介して、一度に1桁ずつドライバによってMODEMへ供給される。電話番号がダイヤルされ且つドライバの制御下において適宜のハンドシェイクルーチンで別のMODEMによって応答されると、電話線を介して2つのMODEMの間に接続が確立される。ダイヤリング及びハンドシェイク動作が実行される期間中、通信プログラムは待機状態であり、接続が確立されたことの表示を探す。その接続を確立すると、MODEMドライバは文字を受信保持レジスタ4

2へ送り、それはMODEMステータスレジスタ52内にステータスフラッグをセットして、ホストプロセッサ10によって受け取られるべきレジスタ42内において情報が得られることを表す。例えばレジスタ52をポーリングすることにより又はレジスタ31において識別される割込を受け取ることにより、通信プログラムがこのフラグが存在することの警告が与えられると、それは該文字を検査し且つデータを送るのに必要なステップを取る。次いで、ホストプロセッサ10は、前述した如く中間レジスタ37、バッファ38、変調器37を介して電話線により送信する為に、該データを送信レジスタ36内に配置即ち供給する。

ホストプロセッサ10は、仮想コントローラとして機能する。換言すると、通信プログラムにとって、それは別のコントローラが、該プログラムによって発生され且つUARTエミュレーションレジスタ33内に配置されたデータを受け取り且つ処理するように見える。然し乍ら、実際には、その情報は、TSRのMODEMドライバの制御下

において、その情報を発生したホストプロセッサによって処理される。従って、既存の通信プログラムとの完全な適合性は維持され、且つMODEMは専用コントローラの必要性なしで動作することが可能である。

前述したことから、保持レジスタ36、42と中間レジスタ37、41との間で文字の転送を制御することによって、電話線を介して転送されるか又は受け取られるデータの各文字をドライバがインターセプト即ち中断することが理解される。このタイプの動作の場合、ドライバが任意の所望の通信プロトコルに従って文字を処理することが可能である。例えば、データをブロック状にバッファ内に格納することが可能であり、且つエラー補正の為にコード化又はデコードさせることが可能である。同様に、効果的な伝送効率を増加させる為にデータを圧縮することも可能である。

これらのタイプのデータ処理動作が所望されない場合、ドライバがデータ文字を中断することは必要ではない。この場合、中間レジスタ37、4

1を除去することが可能であり、且つ保持レジスタ36、42をスイッチ（不図示）によってバッファ38、40へ接続することが可能である。これらのスイッチの開閉は、レジスタ36、42内の文字が、夫々、命令又はデータ文字であるか否かに依存して、ドライバによって制御することが可能である。命令文字は、該スイッチが開成又は閉成している間にドライバによって読み取られ、一方該スイッチが閉成されている場合にのみ、データ文字は直接的に変調器へ送られるか又は復調器から受け取られる。

第4図に示した如く、割込制御を有する中間レジスタ配列を使用することに関連する別の利点は、この構成では、同期から非同期への及び非同期から同期へのデータバッファの必要性を取り除いているという事実である。従来のMODEMにおいて、これらのバッファは、UARTと復調器の出力端と変調器の入力端との間に存在せねばならない。基本的に、該バッファは同期データを非同期データへ変換し、且つその逆へ変換して、ASC

11 適合性及び所定の範囲内のデータ率（速度）を与える。更に詳細に説明すると、同期／非同期バッファは、通常、復調器の出力端子に存在して、ストップビットを失ったフォーマットされた文字を検知する。このバッファは、適宜、該ストップビットを挿入して、該データを所定のライン速度、例えば1,200ボー（baud）MODEM用の1,219ビット／秒でデータを保持する。

同様に、非同期／同期バッファは、典型的には、変調器に対する入力端子において存在する。このバッファは、ホストプロセッサから供給されるデータが早すぎる場合にストップビットを除去して、その際に該変調器に対するビット速度を減少すべく機能する。入力データが遅すぎる場合、1つ又は2個以上のストップビットを挿入してビット速度を良好に固定した範囲内に維持する。

第4図に示したMODEMアーキテクチャにおいて、変調器又は復調器とUARTとの間の通信は該回路にとって内部的であり、従ってデータは非同期装置へ送給されない。従って、厳格なデ

ータ速度をトラッキングしたり又は厳しい非同期インターフェース明細を充足することは必要ではない。動作について説明すると、中間レジスタ37、41とそれらの夫々のシフトレジスタ38、40との間を通過するデータは非同期的であり且つ文字フォーマット化も行わない。データは、TSRMODEMドライバ62を適宜の割合でフラッグすることにより、使用可能な場合に中間レジスタ37及び41内で処理される。同期送信クロック44は、データを中間レジスタ37からレジスタ38内へ並列にシフトする為に使用される。同様に、回復された同期受信データクロックは、データをレジスタ40から中間レジスタ41内へシフトする為に使用される。特定の転送プロトコルの要件を充足する為に、各送信されたワードの始めにスタートビットを及び／又は各ワードの終わりに1つ又は2つのストップビットを付加することが必要な場合がある。これらのビットは、同期送信クロックにより適宜の時間において選択的にイネーブルされるゲート46により変調器に入

るビットストリーム内に挿入させることが可能である。一方、スタートビットに対しシフトレジスタ38の始めにおいて1つのエキストラなステージ（段）を又ストップビットに対し終わりににおいて2個のエキストラなステージを付加し、且つ転送プロトコルにより所要に応じこれらのステージを選択的にイネーブルさせることが可能である。第4図の実施例において使用される割込駆動アプローチの場合、データ速度における変動に対する影響は減少される。何故ならば、割込技術は通常のMODEMデータ速度明細の外側のデータ速度を受け入れることが可能であるからである。更に別の利点として、このアプローチは、動作速度に関係なく、フェーズシフトキー（PSK）又はクアドラチャータン変調（QAM）タイプのMODEMと共に使用することが可能である。

各動作を実行した後にMODEMドライバ62の効率的な出力動作は、それが、不必要にシステム資源を使用することなく、通信プログラムと同時にランすることを可能としている。好適には、

ドライバプログラムは、通信プログラムの正常な機能と干渉することを防止する為に、長期間に渡ってホストプロセッサの制御を維持すべきではない。例えば、MODEMを介してダイリグルーチンを実行する場合、多くの通信プログラムは、所定の文字又は文字列を送ることによってダイリグをキャンセルすることを可能とする。MODEMドライバ62が、全ダイリグルーチン期間中にプロセッサの制御を維持する場合、キャンセル用文字の派生は認識されない。従って、好適実施例においては、MODEMドライバ62は、別の割込、即ちタイマー割込を使用して、それがシステムから効率的に出力されることを可能とし且つ必要な時に制御を移還して、所要の機能を実行する。

コンピュータは周期的な間隔、例えば54ミリ秒毎にタイミング割込要求を発生する。TSRMODEMドライバ62は、タイミング割込ベクトルがそれ自身に向かってポイントすべく割込ベクトルアドレステーブル60内にアドレスをセットする。従って、54秒毎にタイミング割込要求が

発生される時に、ホストプロセサ10は、MODEMドライバ62へジャンプする。このアプローチは、プロセサの制御を維持することなしに、MODEMドライバプログラムがイベント（事象）を同期させることを可能とさせる。該プログラムは、高々、引き続くタイミング割込要求間の期間内にプロセサの制御を奪還するので、その期間内に何もすることが必要でなければ、該プログラムは本システム（方式）から出力することが可能である。

例えば、該プログラムがパルスダイアリング動作を実行していると、MODEMは33ミリ秒の間フック状態とされねばならない。MODEMがフック状態とされると、ドライバ62は、その期間の経過前にそれが該プロセサの制御を奪還するか否かを決定する。タイミング割込が33ミリ秒前に発生することが既知である場合、ドライバ62は次の割込間で該プロセサの制御を放棄することが可能である。然し乍ら、次のタイミング割込が33ミリ秒以上離れている場合、ドライバ62

は、その33ミリ秒が経過する間で該マイクロプロセサの制御を維持する。この為に、ドライバ62は、最も最近の割込からの経過時間、又はその逆に、次の割込への残存時間をカウントするタイマーサブルーチンを有することが可能である。

MODEMドライバがプロセサの制御を放棄する前に、それはその現在或る特定の状態、例えばダイアリング、オンライン、応答、及びハンドシェイク等及びそれが現在存在する状態における特定のポイントの表示を格納する。例えば、MODEMがダイアリング状態にあると、ドライバは、それがトーン発生器をターンオンさせ且つそれをターンオフすることを待機していることの表示を格納する場合がある。該プログラムが次のタイミング割込でシステム（方式）内に再度入ると、それが出た時の状態を注意し且つその特定の状態を取り扱う為のサブルーチンへ分岐する。

更に、ドライバ62は、それがマイクロプロセサの適切な制御を維持することを確保する為にタイミング割込を使用することが可能である。更に

詳細には、幾つかの通信プログラムは、通信割込ベクトルを変化させてそれら自身にポイントさせることが可能である。然し乍ら、適切に機能する為には、TSRMODEMドライバ62は、通信プログラム前にいかなる通信割込要求も中断することが可能でなければならない。このことが発生することを確保する為に、MODEMドライバプログラム62は、それがタイミング割込後に本システム内にエンター即ち入る毎に割込ベクトルアドレステーブル60をチェックすることが可能である。未だに通信割込ベクトルがMODEMドライバ62に対してポイントしている場合、何の動作も行ふことは必要ではない。然し乍ら、ベクトルアドレスが変化されると、ドライバ62は割込ベクトルを再度マップすることが可能であり、従ってそれは通信割込チェーンにおける最初のアドレスである。そのように行う場合、MODEMドライバ62は、テーブル60内にあったアドレス、例えば通信プログラムのアドレスを適宜のレジスタ内に格納することが可能である。プログラム6

2が該割込みによって必要とされる動作を終了すると、それは、該プロセサを以前にテーブル内にあった格納されたアドレスへ参照させることが可能である。

MODEMドライバの前述した動作を、第5A図及び第5B図のフローチャートに詳細に示してある。前述した如く、該プログラム内へのエントリーは、タイマー割込又は通信割込の発生により発生する。タイマー割込が発生すると、プロセサはタイマー割込ルーチン70をコールする。このルーチンの一部として、該プロセサは割込ベクトルアドレステーブル内に格納されたアドレスをチェックして、通信割込ベクトルが未だ該MODEMドライバに対するアドレスへポイントしていることを確保する。そうでない場合、ベクトルアドレスは変化され且つ該テーブル内に存在していたものが該ドライバに対するメモリ空間内の適宜のアドレスに格納される。次いで、メインルーチン72へ動作が継続する。

同様に、通信割込が発生すると、プロセサは最

初に通信割込ルーチン74へ分岐することが可能であり、それにより、それはハウスキーピング即ち通常のルーチン、例えば、レジスタのセット等のルーチンを行い、次いでメインルーチン72へ継続する。

メインルーチンにおいて、プロセサは最初に、出力情報がユーザへ行くことを待機しているか否かを決定する。そうである場合、それは応答サブルーチン76へ分岐し、それは適宜のタイプの出力情報を発生させる。次いで、メインルーチンは、MODEMがオンフックであるか否かを決定する。そうである場合、それは応答サブルーチン78へ分岐して、リングイン信号が存在するか否かを確かめ且つ適切であればそれに対して回答する。最後に、メインルーチンは、ドライバが最後に存在していた特定の状態に対して状態取扱ルーチンへ分岐する。該ルーチンは、ループを有しており、その中において、それは該状態が変化しない限り異なった状態に対して状態取扱ルーチンをコールすることを継続して行う。ドライバの状態が変化

しないと、ドライバは、それが通信プログラムの応対を必要とする通常の割込条件にตอบสนองしていたか否かを決定する。そうであれば、それは通信プログラム58内の適宜のアドレスへプロセサを分岐させる。通信プログラムの応対を必要とする割込条件がない場合、ドライバはイグジットし且つその際に次の割込が発生する間でプロセサの制御を放棄する。

ドライバの種々の状態の例は、MODEMが命令モードにある場合及び命令文字を待機している場合、該文字を受け取る場合、又は該命令を実行する場合に発生する命令状態、MODEMオフフックをとること、ダイアルトーンを待機すること及び番号のダイアリング、応答及びハンドシェイク状態を取り扱うダイアリング状態、及びオンライン状態を包含する。可能な状態の各々に対するドライバの一般的な動作を第5B図に示してある。

プログラムが状態取扱ルーチンにエンターすると、それは、最初に、それがタイマー割込にตอบสนองしているか否かを決定する(決定ステップ80)。

応答していれば、それはローカルのタイマーをアップデートさせるためにルーチン82へ分岐する。次いで、プログラムは可能な興味のあるイベント即ち事象をチェックすべく進行する(ステップ84)。例えば、プログラムがダイアリングモードにあり且つMODEMオフフックとなった場合、それはライン上のダイアルトーンの存在を聴取することが可能である。ステップ86において、プログラムは、興味のあるイベント、例えばダイアルトーンが発生したか否かを決定する。発生していると、それは、そのイベントにตอบสนองする為にルーチン88へ分岐し、即ちダイアル中の番号における最初の桁に対しトーン発生器をターンオンさせる。この応答が開始されると、該プログラムは状態変化が表示されるべきか否かを決定する(ステップ90)。前述した例において、MODEMドライバはダイアルトーンの聴取からダイアル中の番号における最初のトーン又はパルスの発生へ移行している。この場合、状態変化が発生し、従って新しい状態が適宜のステータスレジスタ内に

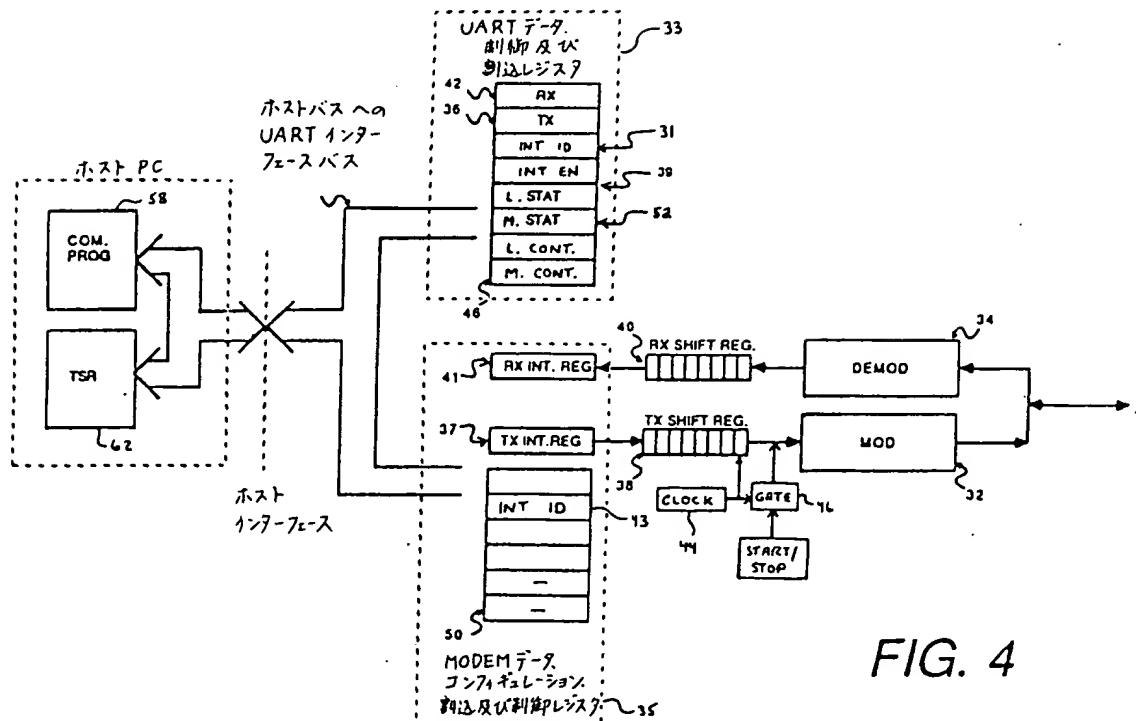
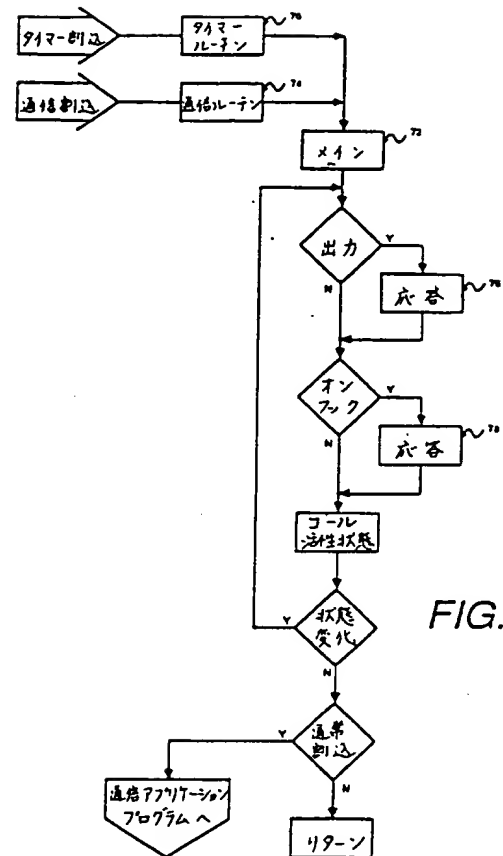
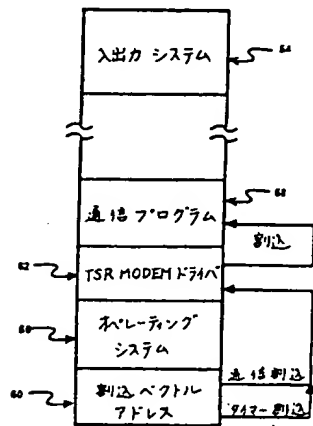
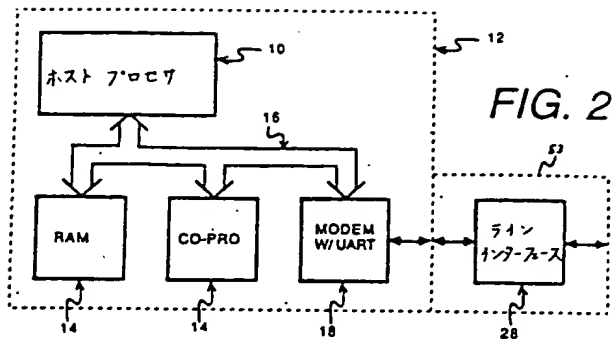
格納され、且つサブルーチンは、状態変化があった表示92と共にメインプログラムへ復帰する。

イベントに対する応答が状態変化とならない場合、又は興味のあるイベントが起らなかった場合、MODEMドライバはその他の興味のあるイベントをチェックすべきか否かを決定する(ステップ94)。そうであれば、前述した手順をこの様な各イベントに対して実施する。最早チェックすべきイベントがなく且つ状態変化もなかった場合、MODEMドライバは継続する。

要するに、本発明は、新しい割込源及び機能を付加しており、従来使用されていたベクトルを再定義して、その際に情報をホストプロセサへ送り返すことによりMODEM内の専用コントローラに対する必要性を排除している。プロセサそれ自身は、通信プログラムが動作するのに必要な制御機能をエミュレートする。専用コントローラを排除することに加えて、必要な制御動作をホストプロセサのメインメモリ14内に格納することによって、処理用ファームウェアも排除されている。







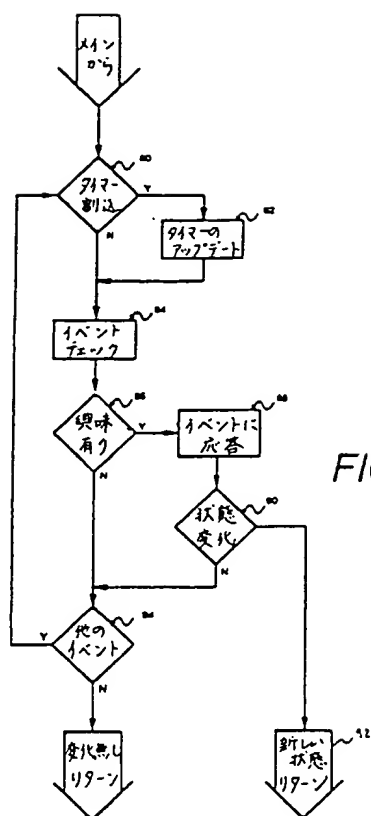


FIG. 5B

第1頁の続き

⑦発明者

アンドリュー ジェ      アメリカ合衆国, カリフォルニア 94301, パロ アル  
 イ. ニコルズ サード      ト, センター ドライブ 525